

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

LISTING OF CLAIMS:

1. (currently amended): A method using a computer system for collecting persistent code coverage data for a computer program, the computer program comprising program source code statements, the method comprising the steps of:

identifying the computer program for which the persistent code coverage data should be collected;

dividing the program source code statements of said computer program into a plurality of code coverage tasks, each of the plurality of code coverage tasks comprising a plurality of basic blocks of code wherein one of the plurality of basic blocks of code comprises a set of consecutive non-control statements with a single entry point and a single exit point and wherein one of the plurality of basic blocks of code comprises a control statement, which comprises a conditional statement, separate from the set of consecutive non-control statements,

wherein any basic block of code which comprises a control statement does not comprise any non-control statements;

generating a persistent unique name for each of the plurality of basic blocks of code of each of the code coverage tasks of said plurality of code coverage tasks;

inserting coverage points into the computer program source code for each of the code coverage tasks to produce an instrumented program;

compiling and linking the instrumented program into a program executable;

identifying a set of test cases from a plurality of test cases to be run for the code coverage data collection purposes;

creating a code coverage database using the code coverage tasks and the identified set of test cases;

running the program executable with a test case from the identified set of test cases and writing the information about the test case and the coverage points that are executed into an output file, until all the test cases have been run; and

processing the information contained in the output file into code coverage data and populating the code coverage database with said code coverage data.

2. (original): The method of Claim 1, wherein generating the persistent unique name for each of the code coverage tasks is done using a unique naming convention.
3. (original): The method of Claim 2, wherein the unique naming convention comprises a computer program module name, a version indicator, and a unique code coverage task identifier.
4. (original): The method of Claim 1, wherein the code coverage database comprises a table, the table comprising a row for each test case in said identified set of test cases and a

column for each code coverage task of said plurality of code coverage tasks, said column comprising an indicator at each row indicating coverage status for said code coverage task.

5. (original): The method of Claim 1, wherein said computer program comprises program source code statements written in a hardware description language.

6. (original): The method of Claim 1 further comprising the steps of:
modifying the computer program to produce a modified version of the computer program source code;

identifying a plurality of new, modified, and deleted code coverage tasks in said modified version of the computer program source code;

generating a persistent unique name for each of the new and modified code coverage tasks of said plurality of new, modified and deleted code coverage tasks;

inserting coverage points into the modified version of the computer program source code for each of the new and modified code coverage tasks to produce an instrumented modified version of the computer program source code;

compiling and linking the instrumented modified version of the computer program source code into a modified program executable;

identifying a new set of test cases from a plurality of test cases to be run for the code coverage data collection purposes on the new and modified code coverage tasks;

altering the code coverage database to accommodate new, modified and deleted code coverage tasks and the new set of test cases, and clearing any code coverage data for the modified code coverage tasks from said code coverage database;

running the modified program executable with a test case from the identified new set of test cases and collecting code coverage data for the new and modified code coverage tasks, until all the test cases have been run; and

updating the code coverage database with the collected code coverage data for the new and modified code coverage tasks;

whereby the previously collected code coverage data for the non-affected code coverage tasks is preserved from a previous version of the computer program to the modified version of said computer program eliminating the need for running the entire test bucket.

7. (original): The method of Claim 6, wherein generating a persistent unique name for each of the modified code coverage tasks is done by changing the version indicator in the names of said modified code coverage tasks.

Claims 8-21 (canceled).

22. (previously presented): The method according to claim 1, wherein the persistent unique name comprises a string having a plurality of components reflecting the code coverage task, wherein only a component of the string of the code coverage task which has been modified is altered while maintaining non-modified components of the string.

23. (new): An apparatus for collecting persistent code coverage data for a program, the persistent code coverage data being stored in a code coverage database associated with the program, the program comprising program source code statements, the apparatus comprising:

a computer system having a data storage device connected thereto, wherein the data storage device stores the code coverage database; and

one or more computer programs executed by the computer system for:

identifying the program for which the code coverage data should be collected;

dividing the program source code statements of said program into a plurality of code coverage tasks, each of the plurality of code coverage tasks comprising a plurality of basic blocks of code wherein one of the plurality of basic blocks of code comprises a set of consecutive non-control statements with a single entry point and a single exit point and wherein one of the plurality of basic blocks of code comprises a control statement, which comprises a conditional statement, separate from the set of consecutive non-control statements,

wherein any basic block of code which comprises a control statement does not comprise any non-control statements;

generating a persistent unique name for each of the code coverage tasks of said plurality of code coverage tasks;

inserting coverage points into the program source code for each of the code coverage tasks to produce an instrumented program;

compiling and linking the instrumented program into a program executable;

identifying a set of test cases from a plurality of test cases to be run for the code coverage data collection purposes;

creating a code coverage database using the code coverage tasks and the identified set of test cases;

running the program executable with a test case from the identified set of test cases and writing the information about the test case and the coverage points that are executed into an output file, until all the test cases have been run; and

processing the information contained in the output file into code coverage data and populating the code coverage database with said code coverage data.

24. (new): The apparatus according to Claim 23, wherein generating a persistent unique name for each of the code coverage tasks is done using a unique naming convention.

25. (new): The apparatus according to Claim 24, wherein the unique naming convention comprises a program module name, a version indicator, a unique code coverage task identifier.

26. (new): The apparatus according to Claim 23, wherein the code coverage database comprises a table, the table comprising a row for each test case in said identified set of test cases and a column for each code coverage task of said plurality of code coverage tasks, said column comprising an indicator at each row indicating coverage status for said code coverage task.

27. (new): The apparatus according to Claim 23, wherein said program comprises program source code statements written in a hardware description language.

28. (new): The apparatus according to Claim 23, wherein the one or more computer programs performed by the computer system further provides for:

modifying the program to produce a modified version of the program source code;

identifying a plurality of new, modified and deleted code coverage tasks in said modified version of the program source code;

generating a persistent unique name for each of the new and modified code coverage tasks of said plurality of new, modified and deleted code coverage tasks;

inserting coverage points into the modified version of the program source code for each of the new and modified code coverage tasks to produce an instrumented modified version of the program source code;

compiling and linking the instrumented modified version of the program source code into a modified program executable;

identifying a new set of test cases from a plurality of test cases to be run for the code coverage data collection purposes on the new and modified code coverage tasks;

altering the code coverage database to accommodate new, modified and deleted code coverage tasks and the new set of test cases, and clearing any code coverage data for the modified code coverage tasks from said code coverage database;

running the modified program executable with a test case from the identified new set of test cases and collecting code coverage data for the new and modified code coverage tasks, until all the test cases have been run; and

updating the code coverage database with the collected code coverage data for the new and modified code coverage tasks;

whereby the previously collected code coverage data for the non-affected code coverage tasks is preserved from a previous version of the program to the modified version of said program eliminating the need for running the entire test bucket.

29. (new): The apparatus according to Claim 28, wherein generating a persistent unique name for each of the modified code coverage tasks is done by changing the version indicator in the names of said modified code coverage tasks.

30. (new): The apparatus according to claim 28, wherein a new persistent unique identifier is generated for any code coverage task, which has been modified or added in the modified source code and

wherein the original persistent unique identifier of all un-modified code coverage tasks is maintained.

31. (new): An article of manufacture comprising a program storage device readable by a computer and tangibly embodying one or more programs of instructions executable by the computer to perform method steps for collecting persistent code coverage data for a computer program, the computer program comprising program source code statements, the method comprising the steps of:

identifying the computer program for which the code coverage data should be collected;

dividing the program source code statements of said computer program into a plurality of code coverage tasks, each of the plurality of code coverage tasks comprising a plurality of basic blocks of code wherein one of the plurality of basic blocks of code comprises a set of consecutive non-control statements with a single entry point and a single exit point and wherein one of the plurality of basic blocks of code comprises a control statement, which comprises a conditional statement, separate from the set of consecutive non-control statements

wherein any basic block of code which comprises a control statement does not comprise any non-control statements;

generating a persistent unique name for each of the code coverage tasks of said plurality of code coverage tasks;

inserting coverage points into the computer program source code for each of the code coverage tasks to produce an instrumented program;

compiling and linking the instrumented program into a program executable;

identifying a set of test cases from a plurality of test cases to be run for the code coverage data collection purposes;

creating a code coverage database using the code coverage tasks and the identified set of test cases;

running the program executable with a test case from the identified set of test cases and writing the information about the test case and the coverage points that are executed into an output file, until all the test cases have been run; and

processing the information contained in the output file into code coverage data and populating the code coverage database with said code coverage data.

32. (new): The article of manufacture according to Claim 31, wherein generating a persistent unique name for each of the code coverage tasks is done using a unique naming convention.

33. (new): The article of manufacture according to Claim 32, wherein the unique naming convention comprises a computer program module name, a version indicator, a unique code coverage task identifier.

34. (new): The article of manufacture according to Claim 31, wherein the code coverage database comprises a table, the table comprising a row for each test case in said identified set of test cases and a column for each code coverage task of said plurality of code coverage tasks, said column comprising an indicator at each row indicating coverage status for said code coverage task.

35. (new): The article of manufacture according to Claim 31, wherein said computer program comprising program source code statements written in a hardware description language.

36. (new): The article of manufacture according to Claim 31 further comprising the steps of:
modifying the computer program to produce a modified version of the computer program source code;

identifying a plurality of new, modified and deleted code coverage tasks in said modified version of the computer program source code;

generating a persistent unique name for the new and modified code coverage tasks of said plurality of new, modified and deleted code coverage tasks;

inserting coverage points into the modified version of the computer program source code for each of the new and modified code coverage tasks to produce an instrumented modified version of the computer program source code;

compiling and linking the instrumented modified version of the computer program source code into a modified program executable;

identifying a new set of test cases from a plurality of test cases that should be run for the code coverage data collection purposes on the new and modified code coverage tasks;

altering the code coverage database to accommodate new, modified and deleted code coverage tasks and the new set of test cases, and clearing any code coverage data for the modified code coverage tasks from said code coverage database;

running the modified program executable with a test case from the identified new set of test cases and collecting code coverage data for the new and modified code coverage tasks, until all the test cases have been run; and

updating the code coverage database with the collected code coverage data for the new and modified code coverage tasks;

whereby the previously collected code coverage data for the non-affected code coverage tasks is preserved from a previous version of the computer program to the modified version of said computer program eliminating the need for running the entire test bucket.

37. (new): The article of manufacture according to Claim 36, wherein generating persistent unique name for the modified code coverage tasks is done by changing the version indicator in the names of said modified code coverage tasks.

38. (new): The article of manufacture according to claim 36, wherein a new persistent unique identifier is generated for any code coverage task, which has been modified or added in the modified source code and

wherein the original persistent unique identifier of all un-modified code coverage tasks is maintained.

39. (new): The method according to claim 6, wherein a new persistent unique identifier is generated for any code coverage task, which has been modified or added in the modified source code and

wherein the original persistent unique identifier of all un-modified code coverage tasks is maintained.